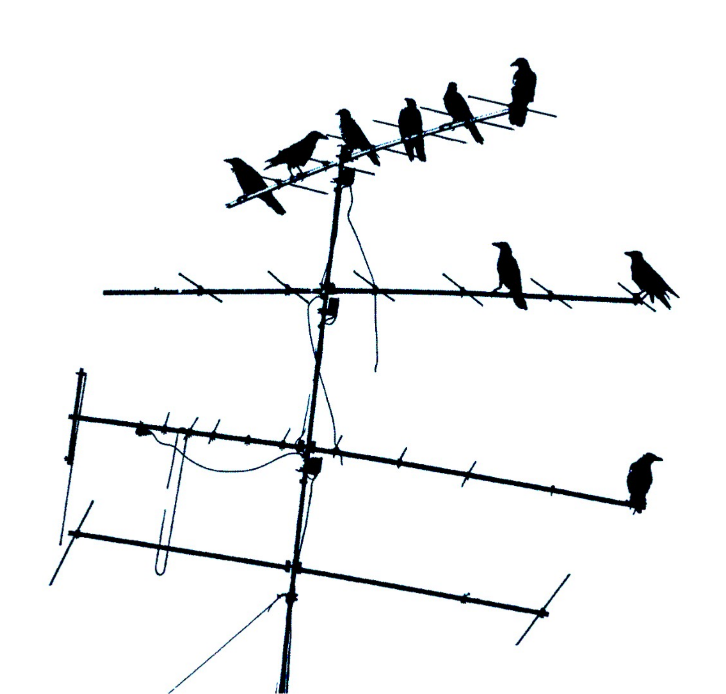
ACTIVISMOS
DEL SOFTWARE LIBRE
COMUNIDAD Y AUTOORGANIZACIÓN

MARGARITA PADILLA



Introducción

El Software Libre es un movimiento social que se preocupa por lograr formas libres, igualitarias, horizontales, comunitarias, autónomas y éticas de utilizar, compartir y crear tecnologías. Es un movimiento que organiza la resistencia contra la privatización del conocimiento.

Los valores del Software Libre, horizontalidad, autonomía, descentralización, cooperación, libre intercambio de saberes, forman parte de la propia identidad de Barrett Comundiad Editorial y del Colectivo "En Las Flores", del que la Editorial forma parte.

A la vez, como ocurre en los grupos hackactivistas de software libre, nuestro pensamiento surge o se vincula fuertemente con nuestra praxis, con nuestra forma de hacer. También, como el Software Libre, aspiramos con nuestras acciones a "hacer comunidad", a la vez que queremos que esas comunidades sean autónomas, igualitarias, libres.

Publicamos aquí una transcripción de la conferencia inaugural y la posterior conversación con el público que la programadora y activista española Margarita Padilla compartió en las 2das jornadas de Género y Software Libre, realizadas en julio de 2019 en la Facultad de Ingeniería y Ciencias Hídricas de la Universidad nacional del Litoral. Dichas Jornadas fueron organizadas por la comunidad de la Tecnicatura Universitaria en Software Libre y el colectivo Trama Tierra.

Para realizarlas se contó con el auspicio del gremio de los docentes de la universidad del litoral (ADUL), los gremios ATE, Amsafé, UPCN, Creative Commons Argentina, Colectivo Libre, Wikimedia Argentina y APUL. Margarita Padilla es una referente destacada del movimiento software libre. Sus ideas se han conocido a

través de libros como "El Kit de lucha en internet"[1], así como a través de entrevistas y de su propio trabajo territorial. Su pensamiento activista sobre las tecnologías, sobre las comunidades, sobre la importancia de lograr soberanía tecnológica, es fundamental no sólo para quienes deseen reflexionar críticamente sobre las tecnologías, sino también para aquellas personas que realizan acciones activistas o militantes.

La exposición de Margarita fue presentada y moderada por Oscar Vallejos, quien enseña e investiga en temas de epistemología e historia de la ciencia y es parte del grupo docente que creó, junto a Martín Bayo, por ese entonces director de la carrera, la Tecnicatura Universitaria en Software Libre de la UNL.

Queremos dejar un agradecimiento especial para Margarita, que nos dio permiso para que realicemos este material, sin pedir nada a cambio y dándonos unas bellas palabras de aliento para seguir adelante.

1 Disponible online: https://traficantes.net/sites/default/files/pdfs/El%20kit%20de%20la%20lucha%20en%20Internet-TdS.pdf

ACTIVISMOS DEL SOFTWARE LIBRE: COMUNIDAD Y AUTOORGANIZACIÓN

Oscar Vallejos [Moderador]: Estamos realmente contentos de que hayas aceptado la invitación. La presentación mía va a ser muy breve porque, como decimos nosotros, conocemos mucho tu trabajo y en cierta medida también tu biografía. Brevemente, lo que nos parecía interesante para conversar no solamente por lo que representa tu trabajo para la comunidad de software libre sino también en relación al perfil, sos una de las pocas mujeres que además de activismo, ha producido una reflexión teórica en muchos soportes. Nosotros te conocimos por primera vez en una entrevista que estaba en Vimeo[2], y cuando te escuchamos flasheamos con lo que vos planteabas, por la manera de plantear tu historia con el software libre, la relación que tramabas entre tu vida, el contexto político de España, y lo que había significado para vos el software libre y las experiencias que tenías en función de eso. Sino también que tenés la virtud de poder hacer activismo, de generar situaciones concretas de trabajo y un alto nivel de reflexión teórica y política, que muchos hemos aprendido entonces de tus textos y de tus intervenciones, que están grabadas y circulando en relación con eso. Así que es para nosotros realmente un gusto tenerte aquí. Si quieres podemos empezar con tu exposición.

[Margarita Padilla]: Buenos días, agradezco mucho esta introducción tan cariñosa que me han hecho, y espero que la conexión nos sea propicia y que pueda plantearles elementos que les ayuden en su reflexión y en sus proyectos.

Quería empezar citando una frase, una expresión de Lawrence Lessig[3] que dice: *"El código es la ley"*. Lessig, como ustedes sa-

2 Varios videos con entrevistas a Margarita pueden encontrarse en Vimeo, por ejemplo: https://vimeo.com/56365525

3 Lawrence Lessig (n. 1961) abogado especialista en derecho informático, crítico del copyright. Pensadores de la "cultura libre" como forma derivada del movimiento software libre.

ben, es un activista en la red, especializado en Derecho Informático y es una persona muy activa en la defensa de la cultura libre. Y es cofundador de las licencias Creative Commons. Él dice que hay tres cosas que regulan la red: una son los mercados, la otra son las leyes y la tercera es el código. Lo que me parece interesante de esta frase es que ve la emergencia de algo nuevo, que podríamos llamarle un espacio –pero da igual como lo llamemos–. Algo nuevo que es distinto del mercado y es distinto del Estado de derecho, que está hecho de código y que tiene una tal capacidad de regulación o tiene un tal poder que deberíamos considerarlo a la altura de los mercados y los Estados.

Lessig es abogado, entonces quizás cuando hace esta afirmación, está jugando deliberadamente con el significado de la palabra código, que se refiere a la vez al código jurídico. Pero yo entiendo que es también aplicable al código de ordenador, al software. Y nuestra tarea sería comprender bien la naturaleza de este nuevo espacio que, a diferencia del mercado –que está fundado en la competencia y en la maximización del beneficio– y a diferencia del Estado de derecho –que estaría fundado en las garantías y los derechos– tiene una naturaleza singular y distinta. Para entender esta naturaleza, desde luego es una tarea creativa, pues cada persona o cada grupo en cada circunstancia la entenderemos… No es algo a estudiar sino que es algo a generar creativamente.

Yo quería proponerles esta línea del tiempo que a mi juicio recoge una historia que sé que es muy conocida por todos ustedes, pero que introduce las claves de los momentos, de los hitos más interesantes en la constitución de este nuevo espacio, que querría repasar muy someramente, muy por arriba, sólo con la finalidad de tener después encima de la mesa todos los componentes o todos los elementos para poder debatir.

Hilo del tiempo

1983
1985
1989
1991
1997
1998
1998

En este hilo del tiempo la primera contribución, la más temprana, es la de Richard Stallman[4] que, como es muy conocido, trabajaba en el Laboratorio de Inteligencia Artificial del MIT. Él vio, observó un proceso en el cual la cultura hacker iba perdiendo, iba retrocediendo ante la presión de las empresas y de la industria del software. Vio cómo sus compañeros empezaron a fundar empresas y cómo en el laboratorio de investigación el software, que antes era un software libre, se iba sustituyendo por software privativo de esta nueva industria. Entonces, para no traicionar a sus principios, constituyó el proyecto GNU[5] para la construcción de un sistema operativo libre. También fundó la Free Software Foundation, institución para coordinar los esfuerzos en ese sentido, inventó el con-

4 Richard Matthew Stallman (n. 1953). También conocido como "rms". Programador y activista. Una de las figuras más visibles del movimiento Software Libre. Autor del manifiesto GNU, fundador de la Free Software Fundation (FSF). Inventó el recurso del *copyleft* como forma de respuesta al copyright. Sus opiniones sobre el caso Epstein le valieron la crítica de gran parte del movimiento Software Libre. Renunció a la FSF en 2019, sólo para ser reincorporado en 2021, algo criticado por varias organizaciones de Software Libre.

5 El proyecto GNU, aunciado por Stallman en 1983, se propuso devolverle a lxs usuarixs el control de su informática, a partir de crear software que sea libre para cualquier uso, para estudiar su código, copiarlo, modificarlo y compartirlo. Las siglas GNU significan "GNU's not Unix!" [GNU no es unix!], haciendo referencia al proyecto de realizar un sistema operativo con todas las prestaciones de un sistema Unix, pero libre.

cepto de *copyleft*[6] e instanció las licencias GPL[7], las licencias libres.

Siguiendo con este hilo del tiempo, llegaríamos a la contribución de Linus Torvalds[8]. Sabemos que en 1991 empezó a trabajar en la construcción del núcleo de un sistema operativo: Linux. Escribió un mítico mensaje en un grupo de noticias, hizo una llamada a colaboraciones y empezó a recibir muchísimas contribuciones de personas de otras partes del mundo. Entonces este hito tiene una grandísima importancia puesto que es la primera vez que se construye un software colaborativo. Obviamente en la historia de la humanidad siempre ha existido y debe existir colaboración, porque sino la humanidad no saldría adelante. Pero lo que podemos ver con la historia de Linux, lo que podemos tener más claramente, es una visión inequívoca de que esta colaboración se produce. O sea, no ya unos procesos más o menos soterrados o naturalizados, sino una contribución organizada colaborativamente y a gran escala.

Otro hito que me parece muy importante en 1998 es cuando Bruce Perens y Eric S. Raymond fundan la Iniciativa del Código

---

6 El *copyleft* es una práctica legal que consiste en el ejercicio del derecho de autor (*copyright* en inglés) con el objetivo de propiciar el libre uso y distribución de una obra, exigiendo que los concesionarios preserven las mismas libertades al distribuir sus copias y derivados.

7 La Licencia Pública General de GNU (más conocida por su nombre en inglés *GNU General Public License* o simplemente sus siglas en inglés GNU GPL) es una licencia de derecho de autor ampliamente usada en el mundo del software libre y código abierto, y garantiza a los usuarios finales (personas, organizaciones, compañías) la libertad de usar, estudiar, compartir (copiar) y modificar el software. Su propósito es doble: declarar que el software cubierto por esta licencia es libre, y protegerlo (mediante una práctica conocida como *copyleft*) de intentos de apropiación que restrinjan esas libertades a nuevos usuarios cada vez que la obra es distribuida, modificada o ampliada.

8 Linux Torvalds (n. 1969) Ingeniero de software conocido por desarrollar el núcleo Linux, siendo todavía hoy uno de sus principales desarrolladores.

Abierto[9]. ¿Por qué si teníamos una Free Software Foundation, una institución para la defensa del software libre, en 1998 ellos consideran que es necesaria otra institución? ¿Cuál es la diferencia? ¿Qué es lo que los motiva? Lo que los motiva es que pasó algo inesperado, completamente inesperado, y es que en 1998 la empresa Netscape liberó –o sea, publicó con una licencia libre– el código de su explorador homónimo, porque estaba perdiendo beneficios. Entonces consideró que quizá publicar el código de su producto como software libre le iba a dar unas grandes ventajas competitivas contra Microsoft e iba a hacer que su producto fuera mucho más popular. De alguna manera, esta liberación de código estaba enmarcada en una batalla competitiva. ¿Cómo es que Netscape llegó a evaluar y tomar esta decisión? Eso fue gracias a la aportación de Eric S. Raymond, a su insistencia en proponerles y en defender que para ellos como empresa era una buena opción.

Entonces si mostramos el Hilo del tiempo, veremos ahí ya todos los hitos rellenados, mostrando el momento en que Raymond publicó su artículo "La catedral y el bazar"[10], en el que expone un nuevo modelo de desarrollo que busca la eficiencia y que es compatible con la empresa, pero que se distancia de los modelos de empresa anteriores o de los modelos de empresa del software privativo, del modelo Microsoft.

---

9 La Open Source Initiative [Iniciativa Código Abierto] se propuso ser una forma menos confrontativa de defender algunas ideas vinculadas previamente al software libre. Haciendo foco sólo en el acceso al código, la iniciativa es considerada por algunxs como una organización que pretende mostrarse "despolitizada" y amigable con las corporaciones.

10 "La catedral y el bazar" es un ensayo sobre el software de código abierto, escrito por el hacktivista Eric S. Raymond en 1997. El ensayo analiza dos modelos de producción de software bien diferenciados. Por un lado, la catedral, que representa el modelo de desarrollo más hermético y vertical característico del Software propietario; y por el otro el bazar, con su dinámica horizontal y "bulliciosa", que caracterizó al desarrollo del kernel Linux y otros proyectos de software libre que se potenciaron con el trabajo comunitario a través de Internet del código abierto.

Hilo del tiempo

1983 – Richard Stallman – Proyecto GNU - (software)

1985 – Richard Stallman – Free Software Foundation FSF- (institución)

1989 – Richard Stallman - Licencia Pública General GNU conocida como GPL - (licencia copyleft)

1991 – Linus Torvalds – Núcleo del sistema operativo Linux – (software colaborativo, comunidad)

1997 - Eric S. Raymond – La catedral y el bazar (modelo de desarrollo, eficiencia, negocio)

1998 - Netscape Communications Corporation (empresa) publica el código fuente de Netscape Communicator (producto)

1998 - Bruce Perens y Eric S. Raymond - Open Source Initiative OSI - (institución)

Quitando las fechas podríamos ver ya expuestos todos los componentes, o al menos los más primigenios, del software libre.

El software por supuesto; distintas instituciones con distintos enfoques, por tanto distintas visiones que se instancian en distintos tipos de licencias, todas ellas libres, pero más o menos permisivas; prácticas colaborativas y de comunidad que han emergido en sus propias instituciones comunitarias y que ya pueden ser observables y estudiables; y modelos de desarrollo de software existentes, que son compatibles con modelos de negocio y de productos.

Componentes

Software

Instituciones diversas, no un único enfoque

Licencias diversas

Prácticas colaborativas, comunidad

Modelo de desarrollo del software eficientes (compatibles con modelos de negocio y de producto)

Lo que a mí juicio es interesante, es pensar cómo estos componentes van produciendo cambios en las relaciones de poder, o en la redistribución del poder. Quería entonces leer a propósito de esto una frase, una expresión de Richard Stallman, que dice:

> *Que las empresas tengan especial influencia en la política significa que la democracia está enferma. El propósito de la democracia es asegurarse de que los ricos no tengan una influencia proporcional a su riqueza. Y si tienen más influencia que tú o que yo, eso significa que la democracia está fallando. Las leyes que obtienen de esta forma no tienen autoridad moral, sino la capacidad de hacer daño.*

Lo que creo que está planteando aquí Richard Stallman, claramente está hablando de política… cómo decirlo. Voy a explicar una anécdota. Recientemente me pidieron que hiciera una unidad didáctica sobre software libre para educadores sociales, para que a su vez los educadores sociales pudieran transmitir esta visión de software libre a chicos y chicas que han sido expulsados –han "fracasado"– del sistema escolar y a los cuales se les quiere dar

una educación tecnológica, con la impresora 3D, con la cortadora láser, robótica y todo eso, y querían que se les explicara el software libre.

Y yo pensé que si a esas personas les empezamos a hablar de licencias y de las cuatro libertades, los vamos a liar a algo que no está en su mundo ¿no? ¿Qué es lo que debe estar a la vez en su mundo y a la vez en el mundo de Richard Stallman?

Y lo que encontré es esta idea de que en muchos casos el derecho de la propiedad debe tener límites. Lo que plantea Richard Stallman es cómo el derecho a la propiedad sobre el software debe tener límites, pero que es algo que podemos llevar a otros contextos sociales, aplicándolo por ejemplo al derecho sobre la vivienda o al derecho sobre el agua, o a cualquier tipo de propiedad, que acumulada y con derechos ilimitados pueda hacer daño. Es muy conocido cómo esa convicción moral de Stallman a menudo se contrapone respecto a la visión más pragmática de Eric Raymond.

Quería entonces leer una cita de Raymond que escribe en su artículo *"La catedral y el bazar"*:

> *A pesar de que la Internet barata es una condición necesaria para que evolucionara el modelo de Linux, no creo que sea en sí misma una condición suficiente. Otros factores vitales fueron el desarrollo de un estilo de liderazgo y el arraigo de hábitos cooperativos, que permiten a los programadores atraer más codesarrolladores y obtener el máximo provecho del medio. Pero, ¿cómo es el estilo de liderazgo y cuáles son estos hábitos? No pueden estar basados en relaciones de poder y, aunque lo fueran, el liderazgo por coerción no produciría los resultados que estamos viendo.*

Dentro de su propio artículo transcribe una cita de Piotr Kropotkin, que es un anarquista ruso, quien dice de sí mismo:

> *Habiendo sido criado en una familia que tenía siervos, me incorporé a la vida activa, como todos los jóvenes de mi época, con una gran confianza en la necesidad de mandar, ordenar, regañar, castigar y cosas semejantes. Pero cuando, en una etapa temprana, tuve que manejar empresas serias y tratar con hombres libres, y cuando cada error podría acarrear serias consecuencias, yo comencé a apreciar la diferencia entre actuar en base en el principio de orden y disciplina y actuar en base al principio del entendimiento. El primero funciona admirablemente en un desfile militar, pero no sirve cuando está involucrada la vida real y el objetivo sólo puede lograrse mediante el esfuerzo serio de muchas voluntades convergentes.*

Respecto a esta cita de Kropotkin, Eric Raymond dice:

> *El "esfuerzo serio de muchas voluntades convergentes" es precisamente lo que todo proyecto estilo Linux requiere; mientras que el "principio de orden y disciplina" es efectivamente imposible de aplicar a los voluntarios del paraíso anarquista que llamamos internet. Para poder trabajar y competir de manera efectiva, los hackers que quieran encabezar proyectos de colaboración deben aprender a reclutar y entusiasmar a las comunidades de interés de un modo vagamente sugerido por el "principio de entendimiento" de Kropotkin. Deben aprender a usar la Ley de Linus.*

Aquí Eric Raymond está hablando directamente de relaciones de poder, pero con un enfoque distinto al de la cita anterior de Stallman, porque Raymond no hablaría tanto de política, sino más de micro-relaciones de poder, relaciones entre las personas. Lo que nos está diciendo es que para lograr grandes proyectos, como por ejemplo los de software colaborativo, necesitamos aplicar este principio de entendimiento. Principio de entendimiento que se pone en evidencia por primera vez con la comunidad Linux, que muestra claramente una idea sorprendente y que el capitalismo o el neoliberalismo niega obstinadamente. Y es que hay mucha gente que quiere colaborar, mucha gente dispuesta a colaborar y que esta colaboración se puede organizar según modelos más o menos organizados –destacando un poco este gradiente de *más o menos*– y que esta organización o autoorganización, según su umbral, da lugares a estas instituciones que llamamos "las comunidades".

El estudio de las comunidades a mí me parece algo apasionante y muy fructífero porque ante estas comunidades de software no estamos ante comunidades deliberativas, tales como serían las comunidades políticas. Recordemos esa diferenciación en tres ejes que hacía Lawrence Lessig: hay Estados, hay mercados y hay código. Entonces el código no es exactamente la lógica política y no es exactamente la lógica de mercado. No estamos ante comunidades deliberativas sino que estamos ante a comunidades productivas, comunidades hacedoras.

Y cuando las comunidades son hacedoras, la misión de la comunidad no es convencer al mundo de que su idea es mejor, porque no son comunidades que propiamente estén trabajando en el terreno de las ideas sino que lo que tienen que hacer es ser, o devenir, convertirse en capaces de convertir su idea en un código mejor, que es un poco diferente. La idea es autoorganizarse para hacer de su idea un código y que ese código sea mejor. Y aquí por código entendería tanto el software producido como los modelos o códigos sociales.

Los modelos bajo los cuales se organiza esa cooperación, según estos principios de entendimiento que cada comunidad instancia un poco a su manera. Así que en este tipo de comunidades, los acuerdos ya no se toman por deliberación y votaciones, como en todos los parlamentos en todos los Estados de derechos del mundo, sino por consenso, muchas veces micro consensos que fluctúan continuamente, en una negociación permanente, en la que todas las partes de la comunidad luchan, trabajan, por hacerse necesarias. Y el mecanismo que tienen esas partes para conseguir ser necesarias es contribuir.

Mientras que en una negociación política la fuerza es la confrontación y, en último extremo, quizás una huelga, en estas comunidades productivas lo que da fuerza para negociar, para conseguir buenos acuerdos, consensos favorables para los intereses de cada una de las partes que forman esta comunidad, es contribuir. ¿Por qué? La relación entre contribución y conocimiento está muy clara: quién más contribuye, más posibilidades tienes de hacer un *fork*. Supongo que es un concepto familiar para ustedes: el *fork* se produce cuando en una comunidad hay una bifurcación, un desacuerdo irreversible e innegociable, la comunidad se divide en dos, normalmente alguien se va, se lleva el código y forma una nueva comunidad con otros códigos sociales.

La gracia no está en llevarse el código, sino en llevarse el conocimiento, poder hacer que ese código mejore, evolucione, etc. Por eso la relación entre conocimiento, contribución y fuerza de negociación, para mí es algo clarísimo. Y estas formas de gobernanza, estos sistemas donde todas las partes luchan y trabajan para hacerse necesarias a base de contribuir, produce equilibrios –no siempre, pero muchas veces– en los juegos de fuerza, en los que todas las partes entienden que alcanzar un consenso va a ser mucho menos costoso que producir un *fork*.

Entonces lo que vemos aquí es que no son comunidades que copian la lógica de la política, en tanto ésta última habitualmente separa el poder político del poder técnico.

Por ejemplo, en un ayuntamiento el poder político está en el ayuntamiento, concejalías, etc. pero hay un cuerpo grandísimo de funcionariado que es estable respecto a ese poder político, y es quien tiene la capacidad técnica de ejecución de esas políticas o de esas gestiones. Está separada la decisión de la ejecución, mientras que en estas comunidades hacedoras la decisión y la capacidad de construcción y reacción de lo que se derive de esa decisión, ha de estar unida. Lo cual no nos lleva a idealizar esas comunidades que están atravesadas de tensiones, de problemas, de relaciones de poder también. Sino más bien, observar su singularidad y ver qué de bueno podríamos encontrar en las propias comunidades de software, y también quizás en otro tipo de proyectos sociales ajenos al software. La potencia de estas comunidades se mueve por tanto en el plano de la cohesión, de un horizonte compartido, que nos haga ver que aunque el *fork* siempre es posible, vamos a intentar evitarlo porque estamos trabajando en la cohesión continuamente y, por otra parte, en esa capacidad productiva.

Quería ilustrar esto con una anécdota: en una contracumbre se quería crear un centro de medios y se quería hacer en ese centro una radio. Se hizo una convocatoria y acudieron "militantes" –activistas políticos forjados en la cultura política– y "hackers" –activistas forjados en esta cultura colaborativa.

Se formó la reunión y fue interminable, larguísima, en la que se iba definiendo el proyecto, tomando los acuerdos, se discutía o se comentaba que tipo de música tenía que sonar en la radio; qué pasaría si nos llegaba una información o un periodista de un periódico de derechas quería usar esa radio para transmitir información; qué pasaría si alguien veíamos que estaba haciendo publicidad de su librería, por ejemplo; o qué pasaría si el lenguaje era sexista; o qué pasaría si esto, qué pasaría si lo otro. En esa reunión tan lar-

guísima, cada media hora los hackers se iban yendo de a uno a la habitación de al lado, hasta que quedaron sólo los militantes. Cuando éstos terminaron su larguísima deliberación, fueron a esa habitación contigua y dijeron: “Ya podemos empezar a hacer la radio, porque ya hemos tomado todos los acuerdos”. Y los hackers respondieron: “Nosotros hace rato que estamos emitiendo”.

¿Qué había pasado ahí? Lo que a mí me parece que había pasado eran dos maneras de trabajar. Y lo que podríamos preguntarnos es por qué para instaurar una comunidad política esos acuerdos debían ser previos y eran tan importantes, mientras que una comunidad productiva sería capaz de empezar a funcionar sin tantos acuerdos. Lo que respondería a eso es que la gente que podía montar una radio, tenía una confianza tal en su conocimiento y en su capacidad hacedora para decir “montamos una radio y la ponemos en marcha, y si no nos gusta, si vienen esas expresiones sexistas, si vienen esos periodistas de la derecha y no sé qué pasa, pues cogeremos nuestro conocimiento y nuestros medios, iremos a otro local y haremos otra radio”. Es decir, “el proyecto no se nos irá de las manos, porque está en nuestras manos hacerlo”. Mientras que en la cultura política hay otra manera de trabajar, que quizás en el debate podamos comentar si estas distintas culturas os parecen familiares, si habéis experimentado algo parecido.

En el siguiente recuadro he intentado expresar algunos valores de la comunidad, intentando plasmar la relación entre conocimiento y comunidad. Porque no debemos olvidar que quien quiera atacar una comunidad, una forma muy clara de hacerlo es desposeerla de su conocimiento. Se puede hacer de muchas maneras: diciendo que ese conocimiento es ilegal, otra manera también habitual es convertir ese conocimiento en obsoleto.

Valores de la comunidad

La inteligencia es colectiva.

El conocimiento crece con la cooperación.

El conocimiento debe ser compartido.

Los desarrollos individuales deben ser devueltos a la comunidad.

El conocimiento no se debe privatizar porque privatizarlo es matar la comunidad.

Si tienes comunidad tendrás libertad. O más aún: sólo dentro de la comunidad eres libre.

Por ejemplo, imaginemos que hay una comunidad de mujeres que mutuamente se facilitan los partos, entonces en esa comunidad hay mujeres y hay un conocimiento sobre los partos que es necesario mantener, actualizar y transmitir. Entonces la comunidad tiene un porqué. Custodia un conocimiento que es importante, en este caso, para la vida. Una manera de destruir esa comunidad es decir que los partos ya no van a ser en casa, sino en el hospital, donde vale un tipo de conocimiento que va a decir que el conocimiento de la comunidad es obsoleto, ya no tiene caso, ya no tiene sentido mantenerlo. Entonces si esas mujeres no encuentran otra manera, otro tipo de conocimiento u otro tipo de hilo conductor de su comunidad, esa comunidad se desmantela porque ya no tiene sentido. Y eso es un proceso que creo que podemos ver continuamente, por parte de ataques del neoliberalismo a todo tipo de comunidades –comunidades de software y de cualquier otro tipo.

Como teníamos una idea de no hacer una intervención larguísima y que hubiera espacio para el debate, si os parece bien terminaría aquí, esperando que podamos armar una conversación interesante.

INTERCAMBIO CON ASISTENTES

[Oscar Vallejos] Voy a empezar con este último ejemplo que pusiste, que es cómo atacar a las comunidades. Me pareció un elemento muy potente, porque efectivamente ocurre esa condición. Y usaste ahí un elemento importante que es la idea de desposeerlas del conocimiento por dos vías: una que es declarar ilegal ese conocimiento y, por otra parte, es la idea de un tipo de conocimiento obsoleto. El ejemplo de saberes sobre el parto es efectivamente muy potente, porque vemos aquí en América Latina que las parteras, quienes sabían sobre eso, fueron atacadas por los médicos de esa manera. Es efectivamente un saber de mujeres que circulaba por la comunidad y que la Universidad y el Hospital los dejó afuera de esa condición. Me parece que es un ejemplo clarísimo de esa forma de atacar, y muy práctica para tener en cuenta cómo operan esos elementos, sobre todo en este contexto que vos llamaste neoliberal, que sería eso: ingresar un tipo de conocimiento que deja afuera a las comunidades de práctica. Entonces quería saber si tenías visto esto en otros espacios, en otros ejemplos.

[Margarita] En el contexto de España las comunidades están ya muy atacadas. Y prácticamente no existen. Quizás es por eso, a raíz del software, cuando tenemos oportunidad de rearmar un tipo de relación social que está completamente devastado. Entonces yo experiencialmente no siento pertenecer a muchas comunidades de conocimiento, porque la comunidad académica no la llamaría una comunidad.

Quizás ustedes estén más cerca todavía de esta cultura comunitaria que no ha sido destruida. Para mí en Europa está completamente devastada. Y es cierto que puede resurgir, por eso para mí la defensa del software libre es tan importante, porque no estamos hablando del software sino de códigos culturales que nos pueden abrir terrenos para soluciones agroecológicas, para nuevas formas

de vida comunitaria, pero que hay que levantar y para las cuales a veces la cultura política presenta ligeros escollos.

No está habiendo estos nuevos modos de colaboración que permiten que la comunidad enseguida se arme. Está la comunidad de hablar y la comunidad de hacer.

Muchas veces las comunidades políticas se basan mucho en el uso de la palabra, la deliberación, los acuerdos. Lo estamos viendo también en las aulas, no puede haber un profesor o una profesora hablando y las personas de brazos cruzados sin hacer nada. No se aprende así, no se aprende escuchando. Entonces para mí el software libre es un modelo no a imitar, pero sí a considerar. De ahí su gran potencia.

[Participante 1] En base a lo que estás diciendo, me parece interesante la distinción que hacés entre las comunidades hacedoras y las comunidades deliberativas. Nosotros venimos trabajando hace mucho tiempo una ley de software libre acá en la provincia de Santa Fe, que justamente le exige al Estado utilizar software libre en todas sus estructuras o en sus tecnologías. Y eso que decís de las comunidades, o el diálogo entre comunidades deliberativas y hacedoras, nos ha costado bastante en esto de pensar con las comunidades locales un activismo sobre impulsar esa ley con más fuerza, con mayor consenso.

Y la verdad que llegarle a las comunidades hacedoras más netas ha sido en disputa con mucha tensión. A veces hemos llegado en mayor medida y a veces no ha habido diálogo. ¿Crees que es posible un diálogo entre ambas comunidades? ¿Cuáles serían los acercamientos en experiencias que nos puedas contar? ¿Cómo se puede establecer un diálogo? Porque me parece que es necesario andar ese camino, de conversar con esas dos lógicas.

[Margarita] ¿En dónde habéis observado más dificultad: en llegar a las comunidades políticas o en llegar a las comunidades hacker?

[Participante 1] Hay como un intermedio porque muchos de los que estamos militando esta legislación, esta ley, estamos también metidos en la militancia técnica del software libre y la soberanía tecnológica. Pero me parece que hay incluso una comunidad más neta del hacer, más hacker y más tecnológica, que es más difícil llegarle.

[Margarita] Las comunidades hacker puras se mantienen en esa ambigüedad, en esa polaridad Richard Stallman - Eric Raymond. Y es raro encontrar comunidades hacker que estén politizadas. La mayoría no lo están.

Podría hablar de la comunidad *Guifi.net*[11], que para mí sería un ejemplo de comunidad hacker que hace política, porque se inaugura claramente con un conflicto, que es cuando las operadoras de telefonía se niegan a dar acceso a los lugares, a poblaciones donde va a haber poco negocio y por tanto la gente que vive allí se queda sin cobertura telefónica o sin acceso a internet. Y eso tiene unas grandísimas consecuencias en la nueva población rural, en todas las personas que quieren vivir en el campo pero sin renunciar a tecnologías.

Y *Guifi.net*... Creo que el truco –digámosle así– es que el momento inaugural constitutivo sea un conflicto, lo cual no puede diseñarse: pasa o no pasa. Si el momento constitutivo es hacer software –por ejemplo, como en el caso de la Wikipedia–, eso puede tener un grandísimo poder de transformación.

Por ejemplo, Wikipedia acabó con el negocio de las enciclopedias, eso está claro. Pero es un conflicto que no tiene una traducción en términos directamente políticos. Mientras en el caso de *Guifi.net* sí, porque está planteando que la comunidad se articula

11 Guifi.net es una red de telecomunicaciones libre, abierta y neutral, mayoritariamente inalámbrica, con más de 32.900 nodos, de los cuales más de 28.800 están operativos (Julio 2015). La mayoría de éstos nodos se encuentran ubicados en Cataluña y la Comunidad Valenciana, aunque se están expandiendo a nuevas zonas en el resto del Mundo. Actualmente, guifi.net es la red libre más extensa de todo el mundo.

sobre un derecho: “cada ciudadano tiene el derecho a la conexión”, que sería lo que aporta la lógica del Estado – garantías y derechos –, pero no lo reivindicamos sino que nos lo autoproporcionamos. Y ahí entra la lógica hacker. Ahora, armar esto es un milagro. Hay que estar muy al acecho para ver por dónde vienen las oportunidades. Yo la verdad no veo muchos ejemplos, pero con uno que haya me basta para ver por ahí la luz.

[Griselda Arias] Hermosa tu presentación. Esto de la comunidad la verdad que lo siento muy así y tu ejemplo de la comunidad de parteras me llegó mucho. En mi experiencia de vida, en esto del parto, sentí esa falta de comunidad. Muy arraigada quedó en mí y quizás eso me trae a la militancia después de siete años de ese momento.

Y lo que también me quedó de recién cuando decías que no está bueno hacer una exposición y que los otros queden en silencio, que está más bueno esto de ir colaborando entre todos y hablando. Y me traía a nuestra educación, ¿no? Yo lo veo mucho en las formas de educar de niños que ya te hacen ser así: está la maestra, yo escucho, y después no puedo opinar ni decir nada. Como que ya desde ahí venimos con esto de cortar la cultura colaborativa, y participativa sobre todo, incluso con castigos.

En mi caso veo cómo los niños influyen en los adultos un montón en todo esto, y por ahí cuando uno los trae a lo que es software libre, creo que todas esas cosas nos ayudan a poder abrir la mente a otra cosa, a los adultos que estamos tan con esta cultura hacedora. No sé si lo ves así, lo menciono como parte de un ejemplito más. Por ahí se puede trabajar más en eso, desde ese lugar.

[Margarita] Creo que la experiencia de estar en un aula con niños y niñas pequeñas se parece mucho a esa experiencia de Kropotkin de trabajar con personas libres. No van a hacerte caso, no van a hacer lo que tú quieras, salvo que las hayas coercionado tanto, destruido tanto, que sólo ya pueden ser sumisas. Eso es lo que pasa ahora mismo en un aula. Hace tres años que voy a un colegio

a enseñar robótica en horario escolar y los talleres míos son desastrosos, porque nunca nada es lo que yo pensé que iba a ser.

Estas experiencias con personas libres, creo que lamentablemente están siendo más por la industria que por las comunidades políticas. Y desde luego en Europa todos los programas innovadores vienen financiados por la industria, por empresas, por Google, por ejemplo. Trayendo estas nuevas prácticas colaborativas a lugares marginados. Y también todos los programas específicos para incorporar a mujeres al campo del software están viniendo por la industria. Creo que están viendo que está inactiva la mitad del talento mundial, cuando podrían tenerla enredada pagando mucho menos dinero. Entonces piensan que estamos perdiendo el tiempo, algo que los Estados y las comunidades políticas no están viendo o no están aprovechando.

Eso para mí es muy preocupante y motivo de tristeza porque tienes que establecer alianzas monstruosas con los financiadores que, por un lado, te están mordiendo, te que están estrujando, pero por otro lado está su visión –que es esta visión tipo Eric Raymond– que ve esto.

En Richard Stallman no tenemos este tipo de textos. No tenemos esta reflexión sobre los estilos de liderazgos, sobre cómo hacer colaboración. Tenemos otro tipo de reflexión, pero no ésta. Y creo que este tipo de reflexión está muy bien acogida por una parte de la industria, no por toda pero por una parte de la industria, y eso para mí es muy preocupante. En los colegios públicos con pocos recursos se están siguiendo los modelos educativos tradicionales, mientras que en los colegios de élite se están siguiendo esos modelos colaborativos. Se está educando a las futuras élites en estos tipos de prácticas hacedoras que al final van a dar una ventaja competitiva y una ventaja social. Ahí yo creo que hay una grandísima reflexión política.

[Oscar Vallejos] Me inquietó o me dio curiosidad ese módulo de didáctica de software libre que estás escribiendo. ¿Por dónde vas? ¿Lo tienes listo, más o menos elaborado? Para conocerlo

[Margarita] Es algo muy modesto. Yo cogí los eventos de la historia de la impresora de Richard Stallman, que supongo que conocen ustedes muy bien: es un grupo de gente que trabaja junto, una impresora que no funciona –un objeto de uso común que no funciona–, un conocimiento sobre cómo repararlo y un límite legal que impide hacerlo. Y esta historia la he traducido, la he ficcionado, en una historia de un grupo de chicos que reciben el regalo de una furgoneta que no funciona, y ellos saben mecánica, pueden repararla, pero se encuentran con unos límites legales que les impiden abrir el motor para repararlo. Esto sirve para que en la unidad didáctica reflexionen sobre si debería haber límites sobre la propiedad intelectual sobre el conocimiento del motor, o no. O si les parece bien que esto sea así. Es algo muy modesto y que os enviaré el enlace en septiembre cuando lo publique.

[Oscar Vallejos] Te preguntaron sobre la relación entre las comunidades políticas y las comunidades hacedoras, como vos las llamás. Como bien venís marcando en varias intervenciones tuyas y comentarios, el problema es que aparecen como lógicas de empresa esas lógicas de colaboración y cooperación que fueron inventadas por las comunidades, y después las empresas se apropian de eso y aparecen con esa lógica. ¿Cómo es entonces esa relación entre las comunidades hacedoras vinculadas con el software libre y esas otras comunidades inventadas por las empresas? Porque uno podría decir que esas son comunidades también inventadas. Google inventa sus propias comunidades. Entonces, vos decías, alguien hace *fork* y se va, pero ahí el trabajo de Google es desposeerte. Que te vayas pero sin nada. ¿Cómo ves esa relación?

[Margarita] La colaboración es posible pero es costosa, sobre todo al principio. Ahora mismo he participado en dos proyectos de software que se quiere pasar a software de comunidad. Uno de

ellos es *Stamapam*, que es un software para localizar en un mapa puntos de comercio social y solidario. Y es un mapa colaborativo pero era un proyecto político, el típico proyecto político en el que el valor se pone en el resultado final de tener el mapa y se pide a una empresa que lo diseñe. Pero con la evolución de esa comunidad ahora ellos quieren que el software sea también colaborativo, especialmente para que pueda ser reutilizado por otras comunidades similares.

He tenido la suerte de ver cómo podría formarse una comunidad, tenemos un software de empresa y queremos pasarlo a colaborativo. Lo que estamos viendo es que al principio hace falta hacer una grandísima inversión, en publicidad, en convocatoria, en dinamización de los encuentros, en trabajar por la inclusión de todas las personas para que ninguna se vaya, en organizar la colaboración. A lo mejor para resolver algo que en término de trabajo de código implica media hora, para que el proceso sea comunitario necesitas tres horas hablando de cuál sería la mejor solución. Vale, empleas tres horas contra media hora. O sea que si me quedo sola, lo resuelvo en un momento, pero no se trata de eso sino de que la solución sea entendida, se vea que es la mejor, sea compartida, etc.

Entonces en términos de rendimiento, si ponemos en un plato de la balanza el tiempo –al final, tiempo y dinero es un poco lo mismo–, o energía invertida respecto a la mejora del código obtenido, el balance es muy pobre. Hay muy poca mejora del código para mucha inversión. Sólo si la inversión es continuada, se mantiene, quizás durante años, al final el balance será positivo. Entonces depende de cuándo tú evalúes pérdidas y ganancias.

En el caso del Linux, Linus Torvalds tuvo de alguna manera mucha suerte. Pero si nosotros hacemos ahora un llamamiento a hacer un sistema operativo, posiblemente no recibamos tantas contribuciones. Él tuvo tanta suerte porque fue el primero y aprovechó una ventana de oportunidad.

Esta colaboración se funda sobre una relación de poder. Quien tiene el poder para hacer una inversión, para mantener una comunidad, es o una iniciativa política que sea muy sólida, muy clara, muy perseverante y que siga un camino durante años, o una empresa, que sabe que esa energía que gasta en formar la comunidad la rentabilizará. Que a la vuelta de seis o siete años, todo serán beneficios. Entonces esto nos lleva a unas comunidades que no son propiamente autoorganizadas, tienen liderazgos muy claros, conviven con modelos de negocio. Por ejemplo, la comunidad WordPress o la comunidad Drupal, que tienen su versión .org y su versión .com que son compatibles, posiblemente permiten colaboraciones, pero las comunidades de software en sí mismas no son antisistema ni mucho menos. Son muy compatibles con el sistema.

Entonces ese esfuerzo, esa molestia, esa especie de acumulación primitiva que tienes que hacer para que después la comunidad vaya rodando. Eso hay que verlo, hay que visionarlo, ver ese horizonte dónde llegarás. Si no lo ves, pues no te pones a hacerlo, porque de primeras no lo vas a conseguir.

Entonces, qué proyectos tenemos. Pues como he dicho antes, Wikipedia por ejemplo, pero ¿quién está detrás de Wikipedia? Pues hay dinero, eso está claro. Quizás no tanto dinero como Google, pero obviamente hay poder, hay capacidad, hay dinero y personas que cuyo trabajo, cuya actividad principal es lanzar eso, porque tienen una idea. Y son modelos muy cercanos a los estilos de liderazgo de Estados Unidos. En la realidad mediterránea no casan bien, porque en el Mediterráneo tenemos otros estilos. Si os fijáis no hay comunidades propiamente mediterráneas, yo no las conozco. Hemos hablado de *Guifi.net*, *Arduino* quizás, pero no tanto. O sea, tenemos problemas.

[Participante 2] Vengo de un recorrido del activismo y la militancia, y en específico la militancia dentro de lo que es la comunicación, los medios de comunicación comunitarios, alternativos, y quería consultarte en relación a lo que mencionabas entre el hacer

y el deliberar. También encuentro que más allá de las diferencias entre esas lógicas de ambas comunidades, me he encontrado en el recorrido últimamente con alguna diferencia entre "lo dado", no sólo entre la lógica interna de cada tipo de comunidad, sino el conflicto entre "lo dado" que ya está dentro de cada una de las comunidades. Entonces quizás del lado de los medios de comunicación comunitarios, que vienen más de una vertiente más deliberativa, de generar consensos y construir cuestiones con sus propias comunidades, veo que es difícil romper la barrera de "si está lo masivo y lo que anda, para qué vamos a ir para el lado del software libre". Entonces lo dado desde ese lado y lo dado también con mucha facilidad como para poder acceder ya sea a determinados servicios, redes sociales o programas para poder automatizar radio, televisión o lo que sea.

Y por otro lado, la cuestión también dentro de lo que son las comunidades hackers, a veces, poder romper determinadas barreras con las discusiones que están más cercanas a la meritocracia. Entonces, por ejemplo, las cuestiones políticas es muy complejo hacerlas entrar, ponerlas sobre la mesa y poder debatirlas. Bueno, estamos en una jornada sobre género donde también es muy difícil a veces poner eso sobre la mesa y discutirlo, cuando la respuesta muchas veces es: "el código está, se puede programar. Acá es una cuestión de sentarse y hacerlo". Y lo político, la cuestión de género y demás es secundario o es más difícil discutirlo.

Entonces mi pregunta es si ves también esa cuestión en ese tipo de comunidades y también si conocés alguna manera o has visto alguna experiencia –mencionaste lo de *Güifi*– donde se puede poner algún tipo de consenso entre esas dos lógicas.

[Margarita] Lamento no poder dar una receta… Me gustaría tenerla, pero no la tengo. Lo más parecido que he visto a esa unión es, por ejemplo, las luchas que hubo en España contra el canon de

los CDs y contra la ley Sinde[12], contra todas las leyes que privatizaban las descargas. Ahí es cuando he visto que se dieran esas alianzas monstruosas entre comunidades más políticas y comunidades más hackers, pero que no quiere decir que vayan a la par sino que sus intereses convergen, lo que es diferente.

Realmente no tengo una gran aportación a tu reflexión, más que mantener viva la llama y formar a personas que puedan circular y ser aceptadas entre los dos mundos, que sean reconocidas porque tienen valía y aportan a los dos mundos, esperando que esas personas encuentren el momento de traducir… De que algunos acontecimientos – que no están en nuestras manos– permitan traducir esas dos lógicas o lenguajes. Te diría que es una tarea de formación y de amor a las dos lógicas.

Una cosa que pasó en una formación a mujeres a la que fui… Pero voy a explicar una cosa que pasó que a mí no me gustó. Era un taller para enseñar a chicas a editar en la Wikipedia y las organizadoras explicaron un poco y la primera práctica que se les propuso a las chicas, que recién están entendiendo qué es la Wikipedia, es abrirse una cuenta y empezar a crear contenidos feministas. ¡No! No es amor a la Wikipedia. Es cierto que en la Wikipedia hay unos sesgos machistas impresionantes, pero a la vez que has de transformar eso, a la vez has de amar el proyecto. A la vez que has de luchar contra el patriarcado interno, no puedes estar ahí sólo para cambiar eso. Sólo entrar como con un búnker, a destruir las lógicas sin construir nada.

---

12 La Ley 2/2011 de Economía Sostenible (LES) es una ley española que tuvo su origen en una iniciativa legislativa aprobada por el gobierno de José Luis Rodríguez Zapatero (PSOE) el 27 de noviembre de 2009. El anteproyecto de ley consistía en un conjunto de medidas que pretendían modernizar la economía española principalmente en tres sectores: financiero, empresarial y medioambiental. Sin embargo, el apartado del anteproyecto que tuvo mayor resonancia entre la opinión pública por la polémica que suscitó, fue la denominada **ley Sinde** (por el apellido de la ministra de Cultura Ángeles González-Sinde Reig), relativa a la regulación de sitios web y la protección de la propiedad intelectual.

Las chicas a los cinco minutos empezaron a recibir ataques de esos machistas que hay en la Wikipedia y no tenían armas para defenderse, no tenían constituida su comunidad de conocimiento y la conclusión de todo eso es “la Wikipedia es un rollo, la Wikipedia es horrible y yo no vuelvo a entrar ahí, porque a la que entro me machacan”. Así es una oportunidad perdida.

Mujeres feministas que son capaces de soportar el patriarcado interno, porque no puedes transformar algo si no estás en contacto con ese algo. Para transformar lo que no te gusta has de estar en contacto con eso que no te gusta. Porque sino no tienes roce y no tienes capacidad de transformación.

Entonces, la formación de personas que podamos, que podáis entender y amar lo que de bueno y de liberador aportan cada una de esas lógicas para transitar entre una y otra buscando la línea de la mayor potencia, para mí eso sería el llamamiento a hacer en horas bajas, cuando no hay nada mejor que hacer. Esperando que acontecimientos, como hemos tenido muchos, como la lucha zapatista, como los atentados de Madrid y Atocha, como un sinfín de momentos políticos en los que ese ponernos manos a la obra toma un grandísimo valor social de liberación y emerge porque ya está construido en sus cimientos. Pero no tengo receta, no tengo un indicativo más allá de este. Yo también sufro estos problemas, estos vacíos.

[Participante 3] Te quería hacer una pregunta que a la vez es un pedido de una receta, por si te animás a darnos alguna sugerencia, más específicamente en relación a los temas de género. Y tiene que ver justamente con los problemas que identificamos y que motiva la realización de estas Jornadas, que tiene que ver con muchas lógicas... Incluso hace poco estuvimos leyendo el informe sobre los labs[13] que sacaron de San Sebastián, donde uno de los lla-

13 El informe “Las voces de la tecnología que resuenan en los labs” recoge los resultados de un trabajo de investigación respecto a lo que pasa en los espacios culturales dedicados a tecnología, que comúnmente se conocen como

mamientos o de los señalamientos es que se repiten, en espacios de software libre, lógicas meritocráticas o patriarcales propias de espacios privativos o de lógicas empresariales.

Por otro lado, vemos que desde esas lógicas empresariales están "robando" la agenda de género, porque ahora todas las empresas tienen capacitación para mujeres, tienen jornadas de mujeres, tienen su área *women* –además nada es en español, todo en inglés– y están como "marketinizando" nuestras luchas. Sin caer o sin seguir esa modalidad de crear un espacio de mujeres, porque pasa lo que vos contabas recién de Wikipedia, porque es agarrar un grupo de mujeres y meterlas en la boca del lobo a que se enfrenten al patriarcado en su máxima expresión. Entonces la pregunta es: ¿cuál sería o cuál pensás que puede ser un camino posible para las comunidades de software libre para ir en pos de reducir la brecha de género, y no ya en una cuestión cuantitativa, sino pensar otras lógicas, otras formas de relacionarnos?

[Margarita] Las comunidades de software libre, en tanto comunidades muy abiertas, están atravesadas por todas estas relaciones de poder que son dominantes en la sociedad.

No sé si conocéis un informe que se hizo sobre los sesgos de género en *GitHub*[14], el repositorio de programas que se usa para recoger contribuciones y mejorar el software. Allí lo que se analizó, lo que se observó, para sorpresa de quienes hacían la investigación es que los aportes de las mujeres son más aceptados que los de los hombres –por tanto se consideran de mayor valía técnica que los de los hombres– siempre y cuando en su perfil no se note que son

espacios makers, de hackeo (hacklabs), laboratorios ciudadanos (Labs), etc., que toma como punto de partida la observación del laboratorio Hirikilabs (Donostia/San Sebastián). Disponible en: http://makusi.tabakalera.eu/ver?id=9563&type=3

14 Informe "Gender Differences and Bias in Open Source: Pull Request Acceptance of Women Versus MenGithub". Disponible en: https://peerj.com/articles/cs-111/

mujeres. Mientras que los de las mujeres que se notan que son mujeres eran menos aceptados que los de los hombres.

También se vio que las mujeres que no se saben que son mujeres, hacen menos aportaciones, o sea trabajan en menos proyectos, contribuyen a menos proyectos, pero las aportaciones son más largas. Digamos que tienen otra relación con su contribución. Siempre he pensado que si a un proyecto –el proyecto que sea– le quieres dar estabilidad, introduce mujeres mayores, porque le van a dar una estabilidad al proyecto, le van a dar una visión a largo plazo, una tranquilidad, una moderación, que va a ser increíblemente buena para lo que quieras hacer. Mientras que posiblemente la gente más joven, con más intensidad, viene y va, y se quemará antes.

Respecto a cómo luchar contra ese patriarcado... En sistemas que son meritocráticos, el reconocimiento entre pares se lo dan entre ellos. Si eres chica, te quedas afuera. Para mí la única receta es mantener grupos de mujeres. Ahora lo que cabría preguntarse sobre esas comunidades de mujeres –por decirlo de alguna manera–, cuál sería el conocimiento, qué tipo de conocimiento vehiculizaría mejor la comunidad. Eso por ahí sería el arte, lo que singularmente en cada situación habría que ver.

Por ejemplo, en un proyecto se planteó esta cuestión y empezamos a hacer cursos de programación sólo para mujeres. Entonces, las primeras oleadas vienen chicas que se están planteando un cambio profesional, o quieren ver qué es eso de la programación. Y en un curso corto de veinte horas, entienden, les fascina y muchas de ellas quieren continuar.

Claro, no se puede montar una universidad sólo para mujeres. Porque ¿qué sería continuar? Aprender mil lenguajes, hay mil tipos de aplicaciones –el software es algo enorme. Entonces ¿qué hacen estas chicas? Se van a esos programas que están dando las empresas, que es como el segundo paso. Entonces las perdemos.

Pero si sabemos ofrecerles entornos y actividades que necesiten, vemos que vuelven.

Porque qué se encuentran allí [en las empresas]: sistemas competitivos, un machaque respecto a la que sirve y la que no, una gran presión por los resultados. Necesitan volver, pero ya no entonces para aprender programación sino para trabajar otro tipo de conocimientos.

Digamos que nuestro arte o habilidad está en observar el proceso e ir aportando a la comunidad, irla sustentando en los conocimientos que la constituyen según sus necesidades en cada momento.

Si nosotras quisiéramos continuar el programa haciendo cursos cada vez más profundos de alguna manera fracasaríamos, porque no tenemos quién los da, con mejores medios y con mejor reclamo para conseguir trabajo. Pero como allí hay un problema, la comunidad sigue siendo necesaria. Entonces vas armando una comunidad de mujeres programadoras, en las que a veces se habla de programación, pero a veces se habla de género, o a veces se ve una película de Donna Haraway, o a veces se lee un libro. Para que cada persona según su necesidad y su momento, pueda entrar y salir, sin tener miedo de que se vayan, porque si se van y no te necesitan es que están confortables con su situación, ya tienen armas para manejarse. Para mí la única receta es seguir manteniendo grupos de mujeres.

No sé si conoces las editatonas en Madrid, donde un grupo de mujeres se reúnen para editar la Wikipedia. Lo que hacen es buscar temáticas, por ejemplo, mujeres deportistas. Entonces, se hace un llamamiento para que venga gente que tenga ese conocimiento para que venga gente que tiene conocimientos sobre mujeres deportistas y esa semana se hace edición sobre mujeres deportistas. Y cuando recibes ataques, hay una comunidad armada y sabes qué artículos concretos son los que defiendes esa semana. Y luego se hará sobre mujeres diseñadoras, y luego sobre antropólogas... Así vas un poco a caballo entre el conocimiento sobre Wikipedia –los

códigos de la comunidad– y el conocimiento sobre la contribución de mujeres en diferentes líneas o disciplinas. Esa mixtura funciona muy bien. Y no es sólo Wikipedia, ni es sólo feminismo.

[Oscar Vallejos] Te agradecemos tu charla, las respuestas y volviendo al inicio en la presentación había algo que me olvidé, que era una especie de poética del pensar y del hacer que vos tenés, y que a nosotros nos resulta profundamente conmovedora e invitante. Te agradecemos mucho.

[Margarita] Yo les agradezco. Se imaginan que después de sesiones como ésta tengo que hacer una cura de ego, para volver a dimensionar mi ser en lo que humildemente es. Les agradezco mucho que me tengan en cuenta y que se acuerden de mí.

ÍNDICE

Introducción → p. 03

Activismos del Software Libre → p. 05

Intercambio con asistentes → p. 19

II Jornadas de Género y Software Libre

4 y 5 de Julio de 2019 – Santa Fe, Argentina

Videoconferencia de apertura: “Activismos del Software Libre: comunidad y autoorganización” por Margarita Padilla



BARRETT

{Comunidad Editorial}

"CUANDO LAS COMUNIDADES SON HACEDORAS, LA MISIÓN DE LA COMUNIDAD NO ES CONVENCER AL MUNDO DE QUE SU IDEA ES MEJOR, PORQUE NO SON COMUNIDADES QUE PROPIAMENTE ESTÉN TRABAJANDO EN EL TERRENO DE LAS IDEAS SINO QUE LO QUE TIENEN QUE HACER ES SER, O DEVENIR, CONVERTIRSE EN CAPACES DE CONVERTIR SU IDEA EN UN CÓDIGO MEJOR, QUE ES UN POCO DIFERENTE.
LA IDEA ES AUTOORGANIZARSE PARA HACER DE SU IDEA UN CÓDIGO Y QUE ESE CÓDIGO SEA MEJOR."

